

# LSIの設計方法および検証方法

## 発明の背景

本発明は、LSIの設計および検証に関するものであり、特に、設計データの機密を保つための技術に属する。

LSIの設計において、回路の中身の機密を保ちたい場合がある。例えば、データの暗号化に関わるLSIでは、その回路の中身が知られると、これにより、暗号化のロジックが解読されてしまうおそれがある。

従来では、このような機密保持を要するLSIの設計は、設計に従事する人を限定したり、あるいは、設計を行う場所を特定したりすることによって、他の人に回路の中身が知られないようにしていた。

近年のLSIの複雑化、大規模化に伴い、1個のLSIの設計には、多数の設計者が携わっている。したがって、設計者や設計場所の限定のみでは、回路の機密を保つのに必ずしも十分ではない。

## 発明の概要

本発明は、LSIの設計において、暗号化処理を採り入れ、従来よりも回路設計データの機密性を高めることを目的とする。

また、暗号化された設計データについて、機密性を保ったまま、検証を実行できるようにすることを目的とする。

具体的には本発明は、LSI設計方法として、与えられた回路設計データに対し、暗号化処理を行うステップを備えたものである。

そして、前記本発明に係るLSI設計方法における暗号化ステップは、前記回路設計データが表す回路の全体または一部を元の回路として回路変換を行い、暗号化回路を生成するステップを備えており、前記回路変換ステップは、前記元の回路と入力数および出力数が同一の少なくとも1つのダミー回路を前記元の回路と並列に配置し、前記元の回路およびダミー回路の出力を並び替える並び替え回路を配置し、前記並び替え回路の出力から、選択信号に従って、前記元の回路の出力数に相当する数の信号を選択するセレクタを配置し、前記暗号化回路を生成

するものとし、かつ、前記選択信号をキー信号とし、前記元の回路の出力と前記セクタの出力とが一致するようなキー信号の値を前記暗号化回路のキーとするものであるのが好ましい。

さらに、前記暗号化ステップは、前記回路変換ステップにおいて用いるダミー回路を生成するステップを備え、前記ダミー回路生成ステップは、前記元の回路に対し変換ルールに従ってダミー回路の候補からなるダミー論理データベースを生成し、前記ダミー論理データベースから出力ルールに従ってダミー回路を選択するものであるのが好ましい。また、前記変換ルールは、論理値の反転、論理演算子の変換、および論理演算子の順序変更のうちの少なくとも1つを含むのが好ましい。また、前記出力ルールは、ランダム選択であるのが好ましい。

また、前記暗号化回路のレイアウトを行うステップを備え、前記レイアウトステップは、前記キー信号の入力信号線を電源およびグランドのいずれにも接続可能なようにレイアウトを行うのが好ましい。さらに、前記レイアウトにおいて、前記キーにしたがって前記キー信号を電源およびグランドのいずれか一方に接続し、前記元の回路のレイアウトを生成するステップを備えるのが好ましい。

また、具体的には本発明は、LSIの検証方法として、リファレンスとなる動作モデルとともに暗号化された回路設計データについて、回路動作の検証を行うステップを備え、前記検証ステップは、前記暗号化された回路設計データを復号化し、実際の設計データと前記動作モデルとを得るステップと、前記実際の設計データについてシミュレーションを実行し、実出力値を得るステップと、前記動作モデルについてシミュレーションを実行し、出力期待値を得るステップと、前記実出力値と前記出力期待値とを比較し、比較結果を出力するステップとを備えたものである。

また、具体的には本発明は、LSIの検証方法として、プロトコル定義とともに暗号化された回路設計データについて、回路動作の検証を行うステップを備え、前記検証ステップは、前記暗号化された回路設計データを復号化し、実際の設計データと前記プロトコル定義とを得るステップと、前記実際の設計データについてシミュレーションを実行し、実出力値を得るステップと、前記実出力値を前記プロトコル定義と比較し、比較結果を出力するステップとを備えたものであ

る。

また、具体的には本発明は、L S I の検証方法として、暗号化された回路設計データをシミュレーションによって検証するステップを備え、前記検証ステップは、不正なアクセスによるシミュレーションを制限するものである。

そして、前記検証ステップは、前記暗号化された回路設計データを復号化し、実際の設計データを得るステップと、前記実際の設計データについてシミュレーションを実行するステップと、前記シミュレーションにおいて、所定の制限情報をカウントするステップと、カウント値が上限値を超えたとき、シミュレーションに制限措置を施すステップとを備えているのが好ましい。さらに、前記所定の制限情報は、シミュレーションの実行ステップ、シミュレーションの実行時間、回路内の特定信号のトグル数、および回路への入力の組み合わせのうちの少なくとも1つを含むのが好ましい。さらに、所定の制限情報をランダムに選択するのが好ましい。

また、前記検証ステップは、前記暗号化された回路設計データを復号化し、実際の設計データを得るステップと、前記実際の設計データについてシミュレーションを実行するステップと、前記シミュレーションにおいて、所定のプロトコル制約条件を違反するか否かをチェックするステップと、違反したとき、シミュレーションに制限措置を施すステップとを備えているのが好ましい。そして、前記所定のプロトコル制約条件は、入力プロトコル、および動作中プロトコルのうちの少なくとも1つを含むのが好ましい。さらに、前記所定のプロトコル制約条件をランダムに選択するのが好ましい。

さらに、前記本発明に係るL S I 検証方法における制限措置は、シミュレーションの停止、実行速度低下および異常実行、シミュレーション結果の非出力、並びに、次ステップに渡すデータまたはキーの非生成のうちの少なくとも1つを含むのが好ましい。

また、具体的には本発明は、L S I 検証方法として、シミュレーションにおける不正アクセスをチェックするチェック回路を含む回路設計データを、暗号化処理するステップと、暗号化された回路設計データを、シミュレーションによって検証するステップとを備え、前記検証ステップは、前記チェック回路を動作させ

て、不正なアクセスによるシミュレーションを制限するものである。

そして、前記チェック回路は、シミュレーションにおいて、所定の制限情報のカウンタ値が上限値を超えたか否かをチェックするのが好ましい。また、前記チェック回路は、シミュレーションにおいて、プロトコル制約条件の違反の有無をチェックするのが好ましい。

また、具体的には本発明は、L S I 設計方法として、与えられた回路設計データから、タイミング情報を抽出するステップと、前記回路設計データを抽出したタイミング情報のみを合わせて所定の変換ルールに従い、暗号化設計データに変換するとともに、少なくとも 1 つの論理ゲートにバッファを付加するステップと、前記暗号化設計データについて付加したバッファのサイズを調整するステップと、バッファサイズ調整後の前記暗号化設計データを、前記所定の変換ルールをキーとして復号化するステップとを備えたものである。

また、具体的には本発明は、L S I 設計方法として、固有 I D の判定回路とともに暗号化された回路設計データについて復号化を行い、実際の設計データと前記固有 I D の判定回路とを得るステップを備え、前記ステップは、入力された固有パラメータによって前記固有 I D の判定回路に正しい値を定義するステップを有するものである。

#### 図面の簡単な説明

図 1 は本願発明者が提案する回路設計スタイルを示す図である。

図 2 は図 1 の回路設計スタイルにおける基本工程のパターンを示す図である。

図 3 ( a ) , ( b ) は本発明に係る処理の流れを示す図である。

図 4 は暗号化処理の一例としての回路変換を示す図である。

図 5 ( a ) ~ ( d ) は図 4 に示す回路変換の具体例を示す図である。

図 6 は図 4 に示す回路変換に用いるダミー回路の生成方法を示す図である。

図 7 ( a ) , ( b ) は復号化処理を説明するための図である。

図 8 は検証処理としての第 1 の判定方法を示す図である。

図 9 は検証処理としての第 2 の判定方法を示す図である。

図 1 0 は検証処理としてのシミュレーション制限方法を示す図である。

図 1 1 ( a ) , ( b ) はチェック回路の一例を示す図である。

図 1 2 ( a ) , ( b ) はチェック回路の一例を示す図である。

図 1 3 はタイミング調整方法を示す図である。

図 1 4 はタイミング調整方法を示す図である。

図 1 5 は暗号化設計データが表す回路構成の一例を示す図である。

図 1 6 は暗号化設計データが表す回路構成の一例を示す図である。

### 発明の詳細な説明

図 1 は本願発明者が提案する回路設計スタイルを示す図である。図 1 に示す設計スタイルでは、機密設計データの中身が見えなくても設計・検証処理 S B が実行可能なように、設計データの暗号化処理 S A および復号化処理 S C が実行される。

暗号化処理 S A では、機密を必要とする回路の設計データ 1 1 に対して暗号化を行い、暗号化設計データ 1 2 と、この暗号化を解除するキー 1 3 とを生成する。暗号化設計データ 1 2 は設計・検証処理 S 2 を実行する利用者に提供される。また、その設計・検証処理 S 2 の必要に応じて、キー 1 3 も併せて提供される。

設計・検証処理 S B では、暗号化設計データ 1 2 について、元の回路の中身が開示されることなく、各種の処理が行われる。復号化処理 S C では、設計・検証処理 S 2 が実行された後の暗号化設計データ 1 4 に対してキー 1 5 を用いて復号化を行い、元の回路の設計データ 1 6 を生成する。

図 2 は図 1 の回路設計スタイルにおける基本工程のパターンを示す図である。同図中、( a ) は暗号化 A、( b ) は暗号データのままの処理 B 1、( c ) は暗号を保ったデータ変換 B 2、( d ) は復号化 C を示している。処理 B 1 と処理 B 2 とは、処理 B 1 では復号化および暗号化は行われず、新たなキーは生成されないが、処理 B 2 では新たな暗号化データが新たなキーとともに生成される点で相違する。

図 3 は本発明に係る処理の流れを示す図であり、図 2 に示す基本工程のパターンを組み合わせたものである。同図中、( a ) に示す処理は、暗号化 A、暗号デ

ータのままの処理 B 1 および復号化 C を組み合わせたものである。例えば、R T L レベルまたはビヘイビアレベルの設計データを暗号化し、この暗号化データについて論理合成を行い、暗号化されたゲートレベルの設計データを出力し、その後ゲートレベルの設計データに復号化する、といった処理がこれに相当する。これにより、論理合成中の設計データの機密を保つことができる。また、暗号化データについて論理合成およびレイアウトを行い、その後マスクデータを復号化するような場合も考えられる。また、(b) に示す処理は、暗号化 A、暗号を保ったデータ変換 B 2 および復号化 C を組み合わせたものである。

以下、各処理の具体例について、順に説明する。

#### <暗号化処理>

##### (回路変換)

図 4 は本発明に係る暗号化処理の一例である回路変換を示す図である。図 4 において、 $f_0$  は元の暗号化されていない回路である。回路  $f_0$  の入力数を  $n$ 、出力数を  $m$  とする。回路  $f_0$  は元の回路の全体を表すものであってもよいし、元の回路の部分回路であってもかまわない。

図 4 に示すように、回路  $f_0$  と並列に、回路  $f_0$  と入力数、出力数が同一の  $(p-1)$  個のダミー回路  $f_1 \sim f_{p-1}$  を配置する。そして、その後段に、並び替え回路 2 1 およびセクタ 2 2 を設ける。並び替え回路 2 1 は回路  $f_0$  の出力と各ダミー回路  $f_1 \sim f_{p-1}$  の出力とを受け、これらの出力を並び替えて出力する。例えば出力  $O_1$  からは、各回路  $f_0 \sim f_{p-1}$  の出力の第 1 ビットを集め並び替えて出力し、出力  $O_2$  からは、各回路  $f_0 \sim f_{p-1}$  の出力の第 2 ビットを集め並び替えて出力する。これにより、並び替え回路 2 1 から、元の回路  $f_0$  の出力数に相当する個数すなわち  $m$  個の、 $p$  ビットの信号が出力される。

セクタ 2 2 は、選択信号 KEY に従って、並び替え回路 2 1 の各出力から 1 ビットずつを選択し、出力する。これにより、回路  $f_0$  と同じ  $m$  個の信号がセクタ 2 2 から出力される。このような回路変換の結果、図 4 に示すような暗号化回路が生成される。

ここで、選択信号 KEY を暗号化回路のキー信号とする。そして、回路  $f_0$  の出力とセクタ 2 2 の出力とが一致するようなキー信号 KEY の値を、暗号化回

路のキーとする。

このような回路変換による暗号化は、変換手順が簡易であり、自動変換が容易である。また、暗号化による遅延増加は、セクタ 22 における遅延のみであり、極めて少ない。

図 5 は図 4 に示す回路変換の具体例を示す図である。いま、元の回路  $f_0$  として、図 5 (a) に示すような 2 入力 2 出力の回路が与えられたとする。この回路  $f_0$  に対して図 5 (b) に示すようなダミー回路  $f_1$  を配置する。図 5 (c) は、回路  $f_0$  をダミー回路  $f_1$  を用いて暗号化した結果を示す図である。さらに、図 5 (c) の回路を合成して、図 5 (d) のような暗号化回路を得る。この回路のキーは  $(0, 1)$  である。

図 6 は図 4 に示す回路変換に用いるダミー回路の生成方法を示す図である。図 6 に示すように、元の回路  $f_0$  に対して、所定の変換ルール 25 に従って、ダミー回路候補を含むダミー論理データベース (DB) 26 を生成する。そして、生成したダミー論理 DB 26 から、所定の出力ルール 27 に従って、任意のダミー回路  $f_1 \sim f_{p-1}$  を出力する。このような生成方法では、変換ルール 25 および出力ルール 27 の設定によって、ダミー回路  $f_1 \sim f_{p-1}$  を柔軟に生成することができるので、自動化処理に適する。

変換ルール 25 の例としては、論理値の反転、論理演算子の変換、論理演算子の順序変更などが挙げられる。論理値の反転では、入力値の反転や出力値の反転の他に、複数ビット信号の一部のビットを反転する方法が考えられる。論理演算子の変換では、AND と OR との変換が考えられる。また、出力ルール 27 の例としては、ランダムに選択する方法や、重複するダミー回路を排除する方法などが考えられる。

#### <復号化処理>

図 4 に示すような回路変換によって得られた暗号化回路をレイアウトする際に、キー信号の入力信号線を、電源およびグランドのいずれにも接続可能なようにレイアウトを行う。これによって、レイアウト工程まで、元の回路の内容について機密を保つことができ、かつ、キーを用いて極めて容易に元の回路の復号化を実現することができる。

図7は本復号化処理を説明するための図であり、同図中、(a)は暗号化回路のレイアウトの一例、(b)は(a)の回路をキーに従って復号化した結果を示す図である。図7(a)に示すように、暗号化回路30に入力されるキー信号KEYの入力信号線31を、電源VDDおよびグランドVSSのいずれにも接続可能なように、レイアウトを行う。そして、図7(b)に示すように、キー(図の例では(0, 1, 0))に従って、キー信号KEYの入力信号線31を電源およびグランドのいずれか一方に接続する(ECO(Engineering Change Order))。これによって、元の回路のレイアウトが復号化される。

#### <設計・検証処理>

##### (判定)

暗号化設計データを、復号化してシミュレーションによる検証を行うとき、シミュレーション結果が正常であるか否かを判定するための期待値が必要になる。ところが、この期待値が外部から見えたとすると、回路の内容がこの期待値から推定可能になり、設計データの機密性が保てない。

そこでここでは、回路設計データを暗号化するとき、シミュレーション結果の期待値となるデータまたは期待値を得る元になるデータを含めて、暗号化を行う。そして、検証処理では、シミュレーション結果と期待値との比較結果に基づき、回路動作が正常か否かを判定する。

図8は本発明に係る検証方法としての第1の判定方法を示す図である。本判定方法では、シミュレーション結果の期待値を得る元になるデータとして、リファレンスとなる動作モデルを用いる。すなわち、図8に示すように、まず、動作モデルとともに暗号化された回路設計データ41をキー52を用いて復号化し(S21)、実際の下位レベルの設計データ43(RTLまたはゲートレベルのネットリスト)と動作モデルの設計データ44とを得る。そして、下位レベルの設計データ43についてシミュレーションを実行し(S22)、実出力値45を得る。また、動作モデルの設計データ44についてシミュレーションを実行し(S23)、出力期待値46を得る。そして、得られた実出力値45と出力期待値46を比較し、各シミュレーション時間において、実出力値45と出力期待値46とが一致しているか否かを判定する(S24)。図8の例では、値が一致してい



るので、結果通知 4 7 として、シミュレーション結果は正常である旨を出力する。

図 9 は本発明に係る検証方法としての第 2 の判定方法を示す図である。本判定方法では、シミュレーション結果の期待値となるデータとして、プロトコル定義を用いる。すなわち、図 9 に示すように、まず、プロトコル定義とともに暗号化された回路設計データ 5 1 をキー 5 2 を用いて復号化し (S 3 1)、設計データ 5 3 と、プロトコル定義 5 4 とを得る。プロトコル定義 5 4 では、設計データ 5 3 が示す回路の入出力および中間ノードの値について、その動作状態が定義されている。そして、設計データ 5 3 についてシミュレーションを実行し (S 3 2)、得られた実出力値 5 5 と、プロトコル定義 5 4 とを比較する (S 3 3)。

なお、第 1 および第 2 の判定方法において、シミュレーション結果が異常である旨の結果が得られたときは、シミュレーション実行結果である実出力値 4 5, 5 5 を、暗号化して出力してもよい。

#### (シミュレーション制限)

暗号化設計データについてシミュレーションを実行し、検証した場合、検証結果出力には、設計データ内の全ての信号線の情報が格納される。多くの入力を与えてシミュレーションを実行し、それから得た検証結果出力を解析すれば、暗号化された回路の内容を知得することが可能になる。

そこで、ここでは、検証結果出力から回路の内容が知得されないように、言い換えると、不正アクセスを監視、防止するために、シミュレーションに制限をかける方法を示す。

図 1 0 は本発明に係る検証方法としてのシミュレーション制限方法を示す図である。図 1 0 に示すように、シミュレーション S 4 2 のチェック S 4 3 では、次のような所定の制限情報 6 4 を、シミュレーションの間カウントする。そして、カウント値が所定の上限値を越えたとき、シミュレーション S 4 2 に制限措置を施す。

- ・シミュレーションの実行ステップ、実行時間
- ・回路内の特定信号のトグル数
- ・回路への入力の組み合わせ

これらの制限情報を、ランダムに選択するようにしてもよい。また、シミュレーションの制限措置としては、次のようなものが考えられる。

- ・シミュレーションの停止、実行速度低下、異常実行
- ・シミュレーション結果の非出力
- ・各信号線のダンプ情報、判定結果などのデータの出力停止
- ・次ステップに渡すデータまたはキーの非生成

また、制限情報 6 4 として、プロトコルの制約条件を設けて、シミュレーションにおいて、このプロトコル制約条件に違反するか否かを判定してもよい。プロトコルの制約条件としては、次のようなものが挙げられる。

- ・回路への入力において許容できるプロトコル（入力プロトコル）
- ・回路内の動作において許容できるプロトコル（動作中プロトコル）

これらのプロトコル制約条件を、ランダムに選択するようにしてもよい。

なお、制限情報 6 4 は、回路設計データの暗号化の際に併せて暗号化し、復号化してもかまわないし、暗号化設計データ 6 1 とは別に与えてもよい。

また、シミュレーションにおいて不正アクセスをチェックするための回路を予め回路設計データに含めておいて、暗号化してもよい。このチェック回路は、シミュレーションのときにのみ動作し、回路設計後は動作がディセイブルされるように、構成する必要がある。

図 1 1 はシミュレーションにおいて不正アクセスをチェックする回路の一例を示す図である。図 1 1 (a) に示すチェック回路は、信号 A の変化回数が所定値（ここでは「8」）を越えたとき、信号 B の値にかかわらず、出力値を“0”に固定するものである。シミュレーション時には、信号 X に“1”→“0”を与え、外部リセットをかける。その後、図 1 1 (b) に示すように、所定の制限情報としての信号 A の変化回数が 8 を越えるまでは出力値は信号 B の値に一致するが、信号 A の変化回数が 8 を越えると、出力値は“0”に固定される。これにより、正しいシミュレーション結果が得られなくなる。回路製造時は、信号 X を“1”に固定し、このチェック回路が動作しないようにする。

図 1 2 はチェック回路の他の例を示す図である。図 1 2 (a) に示すチェック回路は、信号 Y の変化時において信号 A、B がともに“0”であることをプロト

コル制約条件とし、信号Yの変化時に信号A、Bのいずれかが“0”でないときはプロトコル違反として認識し、信号Cの値にかかわらず、出力値を“0”に固定するものである。シミュレーション時には、信号Xに“1”→“0”を与え、外部リセットをかける。その後、図12(b)に示すように、信号Yの変化時において、信号Aと信号Bがともに“0”でないときは、出力値は“0”に固定される。これにより、正しいシミュレーション結果が得られなくなる。回路製造時は、信号Xを“1”、信号Yを“0”に固定し、このチェック回路が動作しないようにする。

#### (タイミング調整)

図13および図14は本発明に係るLSI設計方法としてのタイミング調整方法を示す図である。このタイミング調整方法は、図3(a)に示すフローすなわち、暗号化A→処理B1→復号化Cの流れを含むものである。

図13に示すように、まず、元の回路設計データ71からタイミング情報72を抽出する(S51)。ここでのタイミング情報とは、各論理ゲートから接続先の負荷までの遅延のことをいう。そして、所定の変換ルール73に従って、元の回路の内容が分からないように、かつ、タイミング情報72のみを合わせるように、すなわち各論理ゲートから接続先の負荷までの遅延を変化させないで、暗号化処理を行い(S52)、暗号化設計データ74を生成する。変換ルール73の内容がそのままキー75になる。このとき、図14(a)、(b)に示すように、変換後の論理ゲート78の少なくとも1つについて、タイミング調整のためのバッファ79を付加する。

そして、暗号化設計データ74について、図14(c)に示すように、目標タイミングを満たすように、付加したバッファ79のサイズを調整する(S53)。ここでは、暗号化処理S52で生成したキー75を用いる必要はない。バッファサイズ調整後、キー75すなわち所定の変換ルールを用いて、元の回路77を復号化する(S54)。このとき、図14(d)に示すように、調整されたバッファサイズを基にして、元の論理ゲートに変換する。この結果、目標タイミングが達成される。

このようなタイミング調整方法によると、設計者に対して回路内容を秘匿した

ま、タイミング調整を行わせることができる。

#### (固有 I D 生成)

図 1 5 は暗号化設計データが表す回路構成の一例を示す図である。図 1 5 に示すように、暗号化設計データは、製品の種別を判定するための固有 I D の判定回路としての回路固有 I D レジスタを持っている。この回路固有 I D レジスタに入力される固有 I D の値は、変数で定義されている。また、この固有 I D の変数の値を与える固有パラメータが定義されている。固有パラメータは、実際の固有 I D とは異なる並び順で、定義される。

シミュレーションを行う際に、入力する固有 I D の値を “1 1 0” としても、回路固有 I D レジスタの値とは並びが異なっており、“1 0 1” と新たに定義された固有 I D の値が回路固有 I D と一致して、はじめて、正常に動作する回路に生成される。また、次の設計工程では、入力する固有 I D の値を “0 1 1” としても、新たに “1 0 1” と定義しなおされる。

なお、ここでの説明では、固有 I D および入力する固有 I D の値のビット数を 3 としたが、この値は任意とする。さらに、入力する固有 I D と回路固有 I D レジスタは、並びだけではなく、論理が反転していてもよい。

また、図 1 6 に示す回路では、固有 I D のみではなく、論理回路中の、電源またはグラウンドに固定された全てのノードを全て変数で定義されている。

変数の値は、入力する固有 I D の値と、それ以外の正常に動作するように固定された値が入力され、回路上では、回路固有 I D レジスタ（図中の A，B，C）とそれ以外のレジスタ（図中の D，E）との区別がつかなくなり、容易に固有 I D の値を知ることができなくなる。

なお、上述した各方法は、当該方法を実現するためのプログラムを実行するコンピュータを備えた装置によって実現することができる。また、当該方法を実現するためのプログラムをコンピュータ読み取り可能な記録媒体に記録して、この記録媒体に記録したプログラムをコンピュータに実行させることによって実現することができる。

以上のように本発明によると、暗号化によって、従来よりも回路設計データの機密性を高めることができる。また、暗号化された回路設計データを、機密を保

持したまま、設計・検証させることができる。

## クレーム

### 1. L S I 設計方法は、

与えられた回路設計データに対し、暗号化処理を行うステップを備えている。

### 2. クレーム 1 の L S I 設計方法において、

前記暗号化ステップは、

前記回路設計データが表す回路の全体または一部を元の回路として、回路変換を行い、暗号化回路を生成するステップを備えており、

前記回路変換ステップは、

前記元の回路と入力数および出力数が同一の少なくとも 1 つのダミー回路を、前記元の回路と並列に配置し、

前記元の回路およびダミー回路の出力を並び替える並び替え回路を配置し、

前記並び替え回路の出力から、選択信号に従って、前記元の回路の出力数に相当する数の信号を選択するセレクタを配置し、前記暗号化回路を生成するものであり、かつ、

前記選択信号をキー信号とし、前記元の回路の出力と前記セレクタの出力とが一致するようなキー信号の値を、前記暗号化回路のキーとする。

### 3. クレーム 2 の L S I 設計方法において、

前記暗号化ステップは、前記回路変換ステップにおいて用いるダミー回路を生成するステップを備え、

前記ダミー回路生成ステップは、

前記元の回路に対し、変換ルールに従って、ダミー回路の候補からなるダミー論理データベースを生成し、

前記ダミー論理データベースから、出力ルールに従って、ダミー回路を選択する。

### 4. クレーム 3 の L S I 設計方法において、

前記変換ルールは、論理値の反転、論理演算子の変換、および論理演算子の順

序変更のうちの少なくとも1つを含む。

5. クレーム3のLSI設計方法において、  
前記出力ルールは、ランダム選択である。

6. クレーム2のLSI設計方法において、  
前記暗号化回路のレイアウトを行うステップを備え、  
前記レイアウトステップは、  
前記キー信号の入力信号線を、電源およびグランドのいずれにも接続可能なように、レイアウトを行う。

7. クレーム6のLSI設計方法において、  
前記レイアウトにおいて、前記キーにしたがって前記キー信号を電源およびグランドのいずれか一方に接続し、前記元の回路のレイアウトを生成するステップを備えている。

8. LSIの検証方法は、  
リファレンスとなる動作モデルとともに暗号化された回路設計データについて、回路動作の検証を行うステップを備え、  
前記検証ステップは、  
前記暗号化された回路設計データを復号化し、実際の設計データと、前記動作モデルとを得るステップと、  
前記実際の設計データについてシミュレーションを実行し、実出力値を得るステップと、  
前記動作モデルについてシミュレーションを実行し、出力期待値を得るステップと、  
前記実出力値と前記出力期待値とを比較し、比較結果を出力するステップとを備えている。

9. L S I の検証方法は、

プロトコル定義とともに暗号化された回路設計データについて、回路動作の検証を行うステップを備え、

前記検証ステップは、

前記暗号化された回路設計データを復号化し、実際の設計データと、前記プロトコル定義とを得るステップと、

前記実際の設計データについてシミュレーションを実行し、実出力値を得るステップと、

前記実出力値を、前記プロトコル定義と比較し、比較結果を出力するステップとを備えている。

10. L S I の検証方法は、

暗号化された回路設計データを、シミュレーションによって検証するステップを備え、

前記検証ステップは、不正なアクセスによるシミュレーションを制限する。

11. クレーム10のL S I 検証方法において、

前記検証ステップは、

前記暗号化された回路設計データを復号化し、実際の設計データを得るステップと、

前記実際の設計データについてシミュレーションを実行するステップと、

前記シミュレーションにおいて、所定の制限情報をカウントするステップと、

カウント値が上限値を超えたとき、シミュレーションに制限措置を施すステップとを備えている。

12. クレーム11のL S I 検証方法において、

前記所定の制限情報は、シミュレーションの実行ステップ、シミュレーションの実行時間、回路内の特定信号のトグル数、および回路への入力の組み合わせのうちの少なくとも1つを含む。



13. クレーム12のLSI検証方法において、  
前記所定の制限情報を、ランダムに選択する。

14. クレーム10のLSI検証方法において、  
前記検証ステップは、  
前記暗号化された回路設計データを復号化し、実際の設計データを得るステップと、  
前記実際の設計データについてシミュレーションを実行するステップと、  
前記シミュレーションにおいて、所定のプロトコル制約条件を違反するか否かをチェックするステップと、  
違反したとき、シミュレーションに制限措置を施すステップとを備えている。

15. クレーム14のLSI検証方法において、  
前記所定のプロトコル制約条件は、入力プロトコル、および動作中プロトコルのうちの少なくとも1つを含む。

16. クレーム15のLSI検証方法において、  
前記所定のプロトコル制約条件を、ランダムに選択する。

17. クレーム11または14記載のLSI検証方法において、  
前記制限措置は、シミュレーションの停止、実行速度低下および異常実行、シミュレーション結果の非出力、並びに、次ステップに渡すデータまたはキーの非生成のうちの少なくとも1つを含む。

18. LSI検証方法は、  
シミュレーションにおける不正アクセスをチェックするチェック回路を含む回路設計データを、暗号化処理するステップと、  
暗号化された回路設計データを、シミュレーションによって検証するステップ

とを備え、

前記検証ステップは、前記チェック回路を動作させて、不正なアクセスによるシミュレーションを制限する。

１９． クレーム１８のＬＳＩ検証方法において、

前記チェック回路は、シミュレーションにおいて、所定の制限情報のカウント値が上限値を超えたか否かをチェックする。

２０． クレーム１８のＬＳＩ検証方法において、

前記チェック回路は、シミュレーションにおいて、プロトコル制約条件の違反の有無をチェックする。

２１． ＬＳＩ設計方法は、

与えられた回路設計データから、タイミング情報を抽出するステップと、

前記回路設計データを、抽出したタイミング情報のみを合わせて、所定の変換ルールに従い、暗号化設計データに変換するとともに、少なくとも１つの論理ゲートにバッファを付加するステップと、

前記暗号化設計データについて、付加したバッファのサイズを調整するステップと、

バッファサイズ調整後の前記暗号化設計データを、前記所定の変換ルールをキーとして、復号化するステップとを備えている。

２２． ＬＳＩ設計方法は、

固有ＩＤの判定回路とともに暗号化された回路設計データについて、復号化を行い、実際の設計データと、前記固有ＩＤの判定回路とを得るステップを備え、

前記ステップは、入力された固有パラメータによって、前記固有ＩＤの判定回路に正しい値を定義するステップを有する。

## アブストラクト

L S I の設計において、暗号化処理を採り入れ、従来よりも回路設計データの機密性を高める。暗号化処理では、機密を必要とする回路の設計データに対して暗号化を行い、暗号化設計データおよび暗号化を解除するキーを生成する。暗号化設計データは設計・検証処理を実行する利用者に提供され、必要に応じてキーも併せて提供される。設計・検証処理では、暗号化設計データについて、元の回路の中身が秘匿されたまま各種の処理が行われる。復号化処理では、設計・検証処理が実行された後の暗号化設計データに対して復号化を行い、元の回路設計データを生成する。